Amendments to the Specification:

Please replace paragraphs [0043] and [0048] in their entirety with the following paragraphs, wherein markings are included to show changes made.

[0043] Fig. 7 is a flowchart illustrating a preferred load-balancing algorithm; ~~**and**~~

[0048] In a preferred embodiment, one or more of the stored parameters relate to the asset inventory of each server. The state table may also store other media asset parameters such as whether the asset is a "new release" to help anticipate demand for the asset. The state table additionally may contain parameters concerning the capability of each server such as whether it comprises extended memory or an inline adaptable cache (such as that described in U.S. patent application serial No. **10/609,433, now U.S. patent No. 7,500,055**, entitled **ADAPTABLE CACHE FOR DYNAMIC DIGITAL MEDIA**", filed **June 27,2003**, ~~**(identified by Pennie & Edmonds docket No. 11055-013)**~~ which is hereby incorporated by reference in its entirety for each of its teachings and embodiments), or other unique storage attributes.

*Please add the following paragraphs to the specification, between paragraphs **[0044]** and **[0045]**. The paragraphs presented below are adapted from paragraphs [0057]-[0064] of U.S. Patent Application No. 10/609,433, now U.S. Patent No. 7,500,055, which was incorporated by reference in its entirety at the time of filing the present application. No new matter is added.*

[0044A] Fig. 9 is a flow diagram illustrating proactive caching in a preferred embodiment of the present invention;

[0044B] Fig. 10 is a flow diagram illustrating the use of data that has been proactively cached in a preferred embodiment of the present invention;

[0044C] Fig. 11 is a block diagram illustrating a preferred embodiment of the present invention in which an adaptable cache is integrated with a network interface;

[0044D] Fig. 12 is a flow diagram illustrating the operation of the preferred embodiment of Fig. 11;

[0044E] Fig. 13 is a block diagram illustrating a preferred embodiment of the present invention in which an adaptable cache is integrated with a host bus adapter;

[0044F] Fig. 14 is a flow diagram illustrating the operation of the preferred embodiment of Fig. 13;

[0044G] Fig. 15 is a block diagram illustrating a preferred embodiment of the present invention in which an adaptable cache is connected to a storage-system bus; and

[0044H] Fig. 16 is a flow diagram illustrating the operation of the preferred embodiment of Fig. 15.

*Please add the following paragraphs to the specification, between paragraphs **[0048]** and **[0049]**. The paragraphs presented below are adapted from paragraphs [0080]-[0109] of U.S. Patent Application No. 10/609,433, now U.S. Patent No. 7,500,055, which was incorporated by reference in its entirety at the time of filing the present application. No new matter is added.*

[0048A] In a preferred embodiment, an adaptable cache is adapted to proactively cache resources, and is further adapted to notify potential calling applications and other processes of assets it maintains.

[0048B] Alternatively or in addition, the adaptable cache may be adapted to direct the storage system not to respond to requests for particular assets when the assets are cached in the adaptable cache. Operation of one preferred embodiment for implementing proactive caching and notification is described in connection with Fig. 9.

[0048C] As shown in Fig. 9, in step 901, an adaptable cache monitors an I/O bus for asset requests. These may represent requests for content to be delivered immediately or requests for content to be delivered at a specified later time.

[0048D] When a request is detected, the adaptable cache determines whether a copy of some or all of the asset is stored in a storage medium (step 902). In step 903, the adaptable cache further evaluates the request in accordance with one or more caching rules programmed into a core logic. In a preferred embodiment, these caching rules may take account of parameters maintained by the core logic, such as available capacity in the adaptable cache and the request frequency for the requested asset.

[0048E] On the basis of steps 902-903, the adaptable cache determines whether or not some or all of the requested asset or some related asset should be proactively cached (step 904). If it is determined that some or all of an asset should be proactively cached, the system proceeds to step 905 where the adaptable cache communicates directly with the appropriate storage system or device and transfers all or a portion of the asset into its storage medium.

[0048F] In step 906, the adaptable cache notifies requesting applications and other processes that may require the requested asset of its updated content so that future requests for that asset may be directed to the adaptable cache. These applications/processes, or associated hardware or software may preferably maintain a table that lists assets available from the adaptable cache. Each entity receiving notification from the adaptable cache preferably updates its table appropriately to reflect the current content of the adaptable cache. Processing then proceeds to step 907, described below.

[0048G] If in step 904 it is determined not to cache requested content, the system proceeds directly to step 907 where parameters maintained by the core logic are updated. In a preferred embodiment, such parameters may, for example, include the number of times a particular asset has been requested within a specified amount of time and available capacity within the adaptable cache. Processing then returns to step 901 where the adaptable cache continues to monitor the I/O bus.

[0048H] As will be recognized by those skilled in the art, passive monitoring of a bus by an adaptable cache as described above may be impractical with more modern busses which are often segmented and behave more like networks in which each device sees only traffic specifically addressed to it. Accordingly, in systems comprising such busses, a network interface may be adapted to address each received asset request to both a host processor and to an adaptable cache so that the adaptable cache may monitor traffic between the network interface and the host processor. References to monitoring by the adaptable cache herein should be understood to include both passive monitoring as well as monitoring using such a dual addressing scheme.

[0048I] Alternatively or in addition, an adaptable cache may be adapted to perform interval caching wherein a sorted list of pairs of overlapping requests for the same asset is maintained that identifies pairs of requests with the shortest intervals between their start times. For these pairs, as the first request in the pair is streamed, the streamed content is also cached and then read from cache to serve the second request.

[0048J] One preferred embodiment for operation of a media server comprising an adaptable cache adapted for proactive caching and notification will now be described in connection with Fig. 10. As shown in Fig. 10, in step 1001, a request for an asset is received via a network interface and forwarded to a host processor via an I/O bus. In step 1002, the adaptable cache monitors the I/O bus for such requests, caches appropriate content if warranted under its caching rules, and notifies any requesting applications (including the requesting application running on host processor) of its updated content, as described above in connection with Fig. 9.

[0048K] In step 1003, the host processor determines whether or not the requested asset is available from the adaptable cache, such as by consulting a table that stores current assets maintained by the adaptable cache. If the asset (or some portion of the asset) is available from the adaptable cache, the host processor formulates a request for the asset (or portion thereof) to the adaptable cache (step 1004). In step 1005, the adaptable cache returns the requested asset to the host processor.

[0048L] Otherwise, if the asset is not available from the adaptable cache, the host processor formulates a request for the asset to a storage system (step 1006). The requested asset is read in blocks from a storage device of the storage system and transmitted to the host processor as shown by the iteration of steps 1007-1010. More particularly, for each block, the storage device finds the block on the hard drive (step 1007), reads the block (step 1008), transmits the block (step 1009), and determines whether or not the asset comprises additional blocks (step 1010).

[0048M] Another preferred embodiment for implementing the present system and method is shown in connection with Fig. 11. As in Fig. 10, adaptable cache 600 in Fig. 11 also resides as a device connected to the host side I/O bus 106B. In this embodiment, however, adaptable cache 600 is preferably integrated with network interface 130. The adaptable cache 600 preferably interconnect with the host side I/O bus 106B via interface connection 146. Preferred physical specifications for the adaptable cache in this preferred embodiment comprise:

- the form factor of a network interface card (e.g., a peripheral component interconnect or PC1 card) which may be plugged into an available expansion slot on the host system (e-g., a PC1 slot);
- storage capacity in excess of 1 gigabyte (GB) using replaceable commercially off-the-shelf memory modules ,(such as dual inline memory modules - DIMMs) or fixed memory circuits; and
- conformity to PC1 hot-swap specifications to allow the adaptable cache to be removed from service while the host system is in operation. As noted above, the storage size of the adaptable cache can therefore be altered through a hot-swap without disrupting the operation of the media server.

[0048N] In this preferred embodiment, adaptable cache 600 is programmed to respond directly to asset requests when the requested asset is available in its storage medium. In this way, asset requests may be serviced and delivered from the network interface card, eliminating bus traversals when assets requested by the user reside in the adaptable cache.

[0048O] Operation of the system shown in Fig. 11 will now be described in connection with Fig. 12. In step 1201, an asset request is received at network interface 130. In step 1202, adaptable cache 600 determines if the requested asset is available on the adaptable cache.

[0048P] If the asset is available on the adaptable cache, the request is preferably serviced and delivered to the user from the same card, eliminating bus traversals on buses 106 (step 1203). More specifically, the adaptable cache retrieves the resource from its storage medium, converts it to an appropriate wire format and delivers it to the requesting client.

[0048Q] Otherwise, in step 1204, if the requested resource is not available from the adaptable cache, the request is forwarded to host processor 120 for processing. In step 1205, host processor 120 formulates a request for the asset to storage system 102. In step 1206, the asset is returned to host processor 120, as described above in connection with Fig. 10. In step 1207, host processor 120 converts the asset to an appropriate wire format and delivers it to the client via network interface 130.

[0048R] It should be recognized that the proactive caching and notification described above may also be implemented in this embodiment. Thus, adaptable cache 600 may be adapted to monitor received requests, proactively cache some or all of an asset in accordance with caching rules, and notify one or more applications or processes of content that it is currently storing. Further, the adaptable cache may be adapted to direct the storage system not to respond to requests for particular assets when the assets are cached in the adaptable cache.

[0048S] Another preferred embodiment for implementing the present system and method is shown in Fig. 13. In the embodiment of Fig. 13, adaptable cache 600 is integrated with controller 128, bridging I/O buses 106A, B. In this embodiment, adaptable cache 600 preferably plugs into an expansion slot on the host system and provides multiple standard high-speed interfaces, such as bridging Fibre Channel and PCI I/O interfaces. In this embodiment, preferred physical specifications of the adaptable cache may include:

- the form factor of a peripheral component interconnect (PCI) card;

- storage capacity in excess of 1 gigabyte (GB) using replaceable commercially off-the-shelf memory modules (such as dual inline memory modules - DIMMs) or fixed memory circuits; and
- conformity to PC1 hot-swap specifications to allow the adaptable cache to be removed from service while the host system is in operation. As noted above, the storage size of the adaptable cache can therefore be altered through a hot-swap without disrupting the operation of the media server.

[0048T] Operation of the system shown in Fig. 13 will now be described in connection with Fig. 14. In step 1401, a user request is received at network interface 130. In step 1402, the request is forwarded to host processor 120 via I10 bus 106B. In step 1403, host processor 120 sends a request for the asset to storage system 102 via I/O bus 106B.

[0048U] In step 1404, adaptable cache 600 (integrated with controller 128 in this embodiment) monitors asset requests that traverse I/O buses 106A, B and determines if the requested asset is available on the adaptable cache. In step 1405, if the asset is available on the adaptable cache, it is returned to host processor 120.

[0048V] Otherwise, if the requested resource is unavailable from the adaptable cache, the request is forwarded to storage system I/O bus 106A for delivery to the appropriate storage device 104 where the resource persists (step 1406). In step 1407, the storage device, returns the resource to the requesting application, as described in more detail above. In step 1408, host processor 120 receives the requested resource, as described in more detail above.

[0048W] It should be recognized that the proactive caching and notification described above may also be implemented in this embodiment. Thus, adaptable cache 600 may be adapted to monitor received requests, proactively cache some or all of an asset in accordance with caching rules, and notify one or more applications or processes of content that it is currently storing. Further, the adaptable cache may be adapted to direct the storage system not to respond to requests for particular assets when the assets are cached in the adaptable cache.

[0048X] Yet another preferred embodiment for implementing the present system and method is shown in Fig. 15. In Fig. 15, an adaptable cache 600 resides on storage system 102. Adaptable cache 600 preferably interconnects with storage system I/O bus 106A via a high-speed interface connection 1500. This high-speed interface connection preferably allows adaptable cache 600 to replace or supplement existing hard drive storage devices on storage system 102 (including RAID arrays or JBODS) as long as the system has a compatible receptacle and I/O interface. In this embodiment, preferred physical specifications of the adaptable cache comprise:

- the form factor of a 3.5" hard disk drive with a 1" nominal height;
- dual fibre channel interface utilizing a standard SCA 40-pin connector and operating at transfer rates of either 1 or 2 gigabits per second (Gbps);
- storage capacity in excess of 1 gigabyte (GB) using replaceable commercially off-the-shelf memory modules (such as dual inline memory modules - DIMMs) or fixed memory circuits - this facilitates a lower cost while simultaneously providing the benefit of readily available and quality controlled components; and
- hot-swap capability (the ability to swap or remove the adaptable cache from service while the system is in operation). As noted above, the storage size of the adaptable cache can therefore be altered through a hot-swap without disrupting the operation of the media server.

[0048Y] Operation of the preferred embodiment shown in Fig. 15 will now be described in connection with Fig. 16. In step 1601, a user request is received at network interface 130. In step 1602, the request is forwarded to host processor 120 via I/O bus 106B. In step 1603, host processor 120 sends a request for the asset to storage system 102 via I/O bus 106B.

[0048Z] In step 1604, adaptable cache 600 monitors asset requests that traverse I/O bus 106A and determines if the requested asset is available on the adaptable cache. As noted above, those skilled in the art will recognize that passive monitoring of bus 106B by adaptable cache 600 may be impractical with more modern busses which are often segmented and behave more like networks in which each device sees only traffic specifically addressed

to it. Accordingly, as noted above, in systems comprising such busses, host processor 120 may be adapted to address each received asset request to both storage device 104 and to adaptable cache 600 so that adaptable cache 600 may monitor traffic between host processor 120 and storage device 104.

[0048AA] In step 1605, if the asset is available on the adaptable cache, it is returned to host processor 120. In this case, the adaptable cache or other suitable component in storage system 102 may also preferably be adapted to preclude other storage devices 104 from responding to the request from host processor 120 since such storage device will be unable to retrieve and forward the asset to host processor 120 as efficiently as adaptable cache 600 (step 1606).

[0048AB] Otherwise, if the requested resource is unavailable from the adaptable cache, the request is delivered to the appropriate storage device 104 where the resource persists (step 1607). In step 1608, the storage device returns the resource to the requesting application, as described in more detail above.

[0048AC] It should be recognized that the proactive caching and notification described above may also be implemented in this embodiment. Thus, adaptable cache 600 may be adapted to monitor received requests transmitted via 110 bus 106A, proactively cache some or all of an asset in accordance with caching rules, and notify one or more applications or processes of content that it is currently storing. Alternatively, these caching and monitoring components may be divided. More specifically, a separate monitoring component may be provided on I/O bus 106A to monitor requests as they are received by network interface 130. When appropriate, the monitoring component may instruct adaptable cache 600 (residing, for example, on 110 bus 106A) to retrieve and store some or all of an asset.

[0048AD] It should also be noted that although, in the preferred embodiments described above, system components are linked via PCI buses such as bus 106A, B, these components may alternatively be linked via other bus types or data exchanges such as switched fabric and associated daughtercards.